5   stages capable of reconfiguration to operate in accordance with different data

6   encoding standards; and

7   wherein the tokens provide . reconfiguration information to the standard-

8   dependent processing stages.

1   2.   The multi-standard decoder of claim 1, wherein each of the tokens

2   includes an extension indicator that indicates whether additional words are present.

1   3.   The multi-standard decoder of claim 1, wherein one of the standard-

2   dependent processing stages comprises an inverse quantizer.

1   4.   The multi-standard decoder of claim 3, wherein one of the tokens

2   comprises a first QUANT_TABLE token.

1   5.   The multi-standard decoder of claim 4, wherein the inverse quantizer

2   recognizes the first QUANT_TABLE token and, responsive to a first state of the

3   extension indicator in a first word of the first QUANT_TABLE token, generates a

4   second QUANT_TABLE token to be conveyed to another of the processing stages.

1   6.   The multi-standard decoder of claim 5, wherein the second

2   QUANT_TABLE token includes quantization table values.

1   7.   The multi-standard decoder of claim 4, wherein responsive to a second

2   state of the extension indicator of the first word of the QUANT_TABLE token, the

3   inverse quantizer installs a quantization table of the first QUANT_TABLE token in a

4   memory.

1   8.   A method of decoding a data stream of data encoded by different

2   standards comprising:

3   receiving tokens at a standard-dependent processor, the standard-dependent

4   processor capable of reconfiguration to operate in accordance with the different

5   standards; and

6   reconfiguring for standard-dependent processing in response to the received

7   tokens.

1   9.   The method of claim 8, wherein each token includes an extension

2   indicator that indicates whether additional words are present and has a first and a

3   second state to indicate reconfiguration information.

1   10.   The method of claim 8, wherein one of the conveyed tokens is a first

2   QUANT_TABLE token, and further comprising:

3   recognizing the first QUANT_TABLE token; and

4   responsive to the first state of the extension indicator in a first word of

5   the first QUANT_TABLE token, generating a second QUANT_TABLE token to be

6   conveyed to another processor.

1   11.   The method of claim 7, wherein the second QUANT_TABLE token

2   includes quantization table values to be used by the another processor.

1   12.   The method of claim 9, further comprising:

2   responsive to a second state of the extension indicator of the first word of the

3   QUANT_TABLE token, installing a quantization table of the first QUANT_TABLE

4   token in memory.

1   13.   A system comprising: